

PHP'ye Giriş

Hidayet Doğan

Türkiye PHP Grubu

hdogan@hido.net

Özet: PHP genellikle web programlama için kullanılan ve dünyada geniş bir kullanıma sahip, P dilleri ailesinden bir betik dilidir. Derleyicisi olmayan PHP betikleri çalışması için bir yorumlayıcıya ihtiyaç duyar. Yazım kuralları kendine has olmakla beraber C ve Perl dillerine benzemektedir. Öğrenimi kolay ve programlamayı hızlandıran basit bir yapıya sahiptir. Genellikle web programlama için kullanıldığından ötürü HTML içine gömülebilir bir yapıdadır. Dil yapısı ve özellikleri ilk sürümlerinden sonra hızlıca gelişerek nesneye dayalı bir yapıya da bürünmüştür. Birçok yardımcı fonksiyon, eklenti ve kütüphanesi bulunan PHP dili bilindik işletim sistemleri ve web sunucuları üzerinde sorunsuz ve kararklı bir şekilde çalışabilmektedir. Bilindik ve milyonlarca sitenin hazırlanmasında kullanılmış, üzerine birçok kitap yazılmış ve çevirimiçi, çevirimdışı kaynağa sahip olan, gelişimine ve güncellenmesine devam eden bir dildir.

Anahtar Sözcükler: php, web, programlama, betik

1. Tarihi

1995 yılında Rasmus Lerdorf isimli yazılımcı tarafından basit Perl betikleri seti olarak kendi özgeçmişine web üzerinden yapılan erişimleri takip etmek amaçlı oluşturulmuştur. Rasmus Lerdorf bu betik setine ilk başta "Personal Home Page Tools" (Kişisel Sayfa Araçları) ismini vermiştir. İleriki zamanlarda bu set içerisine gereksinim duyduğu özellikleri C dili ile tekrar uyarlayarak PHP/FI (PHP Form Interpreter) ismi ile kaynak kodunu yayınlamıştır. PHP/FI ismi içindeki FI (Form Interpreter) kısaltması web formlarının yorumlanmasından ötürü verilmiş bir isimdir. Bu ilk sürüm ile birlikte HTML içine gömülebilir bir dil ortaya çıkmıştır.

1997 yılında PHP/FI kaynak kodları tekrar yazılmış ve PHP sürüm 3'e ait deneme sürümleri yayınlanmıştır.

1998 yılında PHP sürüm 3 yayınlanmış ve şu anki PHP sürümlerine yakın ve birçok yeni özelliği, eklentiyi barındıran bir sürümdür.

Andi Gutmans ve Zeev Suraski isimli iki yazılımcı PHP sürüm 3 kaynak kodlarını tekrar yazarak, daha performanslı ve modüler

bir yapıya sahip yeni bir altyapı oluşturmuş ve buna "Zend Engine" ismini vermiştir. 2000 yılında yeni altyapıyı kullanan PHP sürüm 4 yayınlanmıştır. PHP sürüm 4 ile birlikte daha geniş web sunucu desteği ve nesneye dayalı programlama desteği getirilmiştir.

Andi Gutmans ve Zeev Suraski, İsrail merkezli "Zend" isimli bir firma kurarak, Zend Studio, Zend Server ve Zend Guard gibi lisanslı PHP ürünleri üretmektedirler ve PHP için sertifikalandırma (Zend Certification) hizmeti vermektedirler. [1]

2004 yılının ortalarında Zend Engine sürüm 2 ile birlikte PHP sürüm 5 yayınlanmış ve yeniden tasarlanan nesneye dayalı model ve eklenti ile beraber gelmiştir.

Şu anda PHP sürüm 5.4 üzerinde çalışmakta olan geliştirme ekibi nesneye dayalı model üzerine ek geliştirmeler ve daha önce oraya çıkabilecek güvenlik açıkları ile ilgili katkı değişiklikler yapmaktadırlar. [2]

2. Kurulum ve Çalıştırma

Özellikle PHP sürüm 4 ile birlikte birçok web sunucusu ve işletim sistemi üzerine

çalışabilen PHP dili; Apache, IIS, Lighttpd ve Nginx gibi büyük kullanım oranına sahip web sunucuları üzerinde basitçe kurulabilmekte ve kararlı bir şekilde çalışmaktadır.

Büyük bir oranla Linux işletim sistemi üzerinde kurulan ve çalışan PHP; Microsoft Windows, Microsoft Windows Server ailesi üzerine de basitçe kurabilen ve yeni Intel tabanlı Mac OS işletim sistemi üzerinde kurulu gelen bir dildir.

Linux dağıtımları için hazır kurulum paketleri bulunmaktadır ve Microsoft Windows üzerinde, IIS web sunucusu için de Zend ve Microsoft ortaklığı ile hazırlanmış hazır bir kurulum paketi bulunmaktadır. [3][4] Özellikle geliştirme süreci için kurulum sürecini hızlandıran ve Windows işletim sistemi için hazırlanmış üçüncü parti yazılımlar da mevcuttur.

Kaynak kodu açık dağıtılan PHP, kaynaktan derlenerek te kurulabilir.

PHP dili komut satırından veya web sunucusu üzerinde yorumlanarak çalıştırılabilir. Komut satırında "php" komutu ile çalıştırılır. Web sunucuları üzerinde ise bir yorumlayıcı (PHP interpreter) olarak görev alır. Bu yorumlayıcı web sunucusu tarafından çalıştırılır ve yorumlayıcının verdiği sonuç ziyaretçinin tarayıcısına gönderilir.

3. Yazım Kuralları ve Dilin Özellikleri

PHP ile hazırlanmış bir betik dosyası PHP yorumlayıcısı tarafından, diğer betik dillerindeki gibi üstten alta ve soldan sağa doğru satırlar yorumlanır ve çalıştırır. Yorumlanma sırasında oluşan yazım hataları bildirilir. Genellikle web sunucusu üzerinde çalışan PHP betikleri ".php" uzantısını alır. PHP olarak yorumlanacak dosya uzantısı web sunucusu ayarlarına göre değişebilir veya değiştirilebilir.

Yazım kuralları oldukça basit ve C, Perl gibi dillere benzerliği olan PHP dili herhangi bir

metin düzenleyici kullanılarak oluşturulabilir. [5] Kısaca yazım kuralları şöyledir:

- HTML içine gömülebilir bir yapıya sahip olduğundan betik içinde PHP kodları `<?php` ayracı ile `?>` ayracı arasına alınır. Eğer betikten sonra herhangi bir HTML kodu (ya da ekrana bir çıktı) verilmeyecekse kapatma ayracı, yani `?>` ayracı kullanılmayabilir.
- Değişken türlerini belirtmeye gerek yoktur.
- Değişken isimleri \$ (dolar) işareti ile başlar.
- Değişken isimleri - (tire) işareti içeremez.
- Değişken isimleri rakam ile başlayamaz.
- Kod blokları { ve } sembolleri arasına alınır.
- Kod içinde bulunan yorumlar /* ve */ ayracıları arasına alınır veya yorum olarak yazılacak satır başına // veya # işaretleri eklenir.

Birçok dilde benzer olarak kullanılan ve mantıksal işlemlerin yapıldığı; "if", "else", "for", "foreach", "do", "while", "return", "break", "continue" gibi komutları bulunmaktadır. PHP sürüm 5.3 ile birlikte "goto" komutu da eklenmiştir. Komutlar arasında PHP diline özgün olan "echo" komutu bulunmaktadır ve ekrana çıktı vermek için kullanılmaktadır.

Bunlara ek olarak diğer dillerde de görebileceğimiz "include" (başka bir dosyayı dahil etme) komutu ve ek olarak dahil edilecek dosyanın varlığını kontrol eden "require" komutu bulunmaktadır.

Prosedürel dillerde aşına olunan fonksiyonlar "function" komutu ile tanımlanırlar.

Şekil 1'de yazım kurallarını anlamak açısından örnek bir PHP betiği görülebilir.

```

1 <?php
2 $degisken = 1;
3
4 if ($degisken == 1) {
5     echo "Değişkene ait değer: 1";
6 }
7 else {
8     echo "Değişkene ait değer farklı.";
9 }
10
11 echo "1'den 10'a kadar rakamlar: ";
12
13 for ($i = 1; $i <= 10; $i++) {
14     echo $i . " ";
15 }
16 ?>

```

Şekil 1: Örnek PHP betiği.

Şekil 2'de HTML içine gömülebilirliği gösteren örnek bir PHP betiği görülebilir.

```

1 <html>
2 <head>
3     <title>HTML içine gömülebilir</title>
4 </head>
5 <body>
6     <h1>HTML içine gömülebilir</h1>
7     <p>
8     <?php
9     $degisken = 1;
10
11     if ($degisken == 1) {
12         echo "Değişkene ait değer: <strong>1</strong><br>";
13     }
14     else {
15         echo "Değişkene ait değer farklı.<br>";
16     }
17
18     echo "1'den 10'a kadar rakamlar: <ul>";
19
20     for ($i = 1; $i <= 10; $i++) {
21         echo "<li>" . $i . "</li>";
22     }
23
24     echo "</ul>";
25 ?>
26 </p>
27 </body>
28 </html>

```

Şekil 2: Örnek PHP betiği: HTML içine gömülebilirlik.

Aritmetik işlemleri, koşulları ve karşılaştırmaları ifade etmek için kullanılan operatörler hepimizin okulda öğrenmiş olduğu aritmetik sembollere benzer şekilde kullanılır. Bunlar;

- Toplama işlemi için + (artı) operatörü,
- Çıkarma işlemi için - (eksi/tire) operatörü,
- Çarpma işlemi için * (yıldız) operatörü,
- Bölme işlemi için / (eğik çizgi) operatörü.

Bunlara ek olarak modülüs ismini verdiğimiz ve % (yüzde) sembolü ile ifade edilen bir operatör bulunmaktadır. Bu operatör bölüm işleminden sonra kalan kısmını ifade etmek için kullanılır.

Aritmetik işlemlerde yapılacak işlemin sırası cebir kurallarında da olduğu gibi önce parantez içlerinde bulunan işlemler gerçekleştirilir. Çarpma ve bölme işlemi de öncelikli olarak gerçekleştirilir.

Değişkenlere herhangi bir değer atama işlemi için = (eşit) operatörü kullanılır.

Şekil 3'te aritmetik ve atama işlemleri için örnek bir PHP betiği görülebilir.

```

1 <?php
2 $degisken1 = 5;
3 $degisken2 = 7;
4
5 $toplama_sonucu = $degisken1 + $degisken2;
6 $cikarma_sonucu = $degisken2 - $degisken1;
7 $carpma_sonucu = $degisken1 * $degisken2;
8 $bolme_sonucu = $degisken2 / $degisken1;
9 $bolmeden_kalan = $degisken2 % $degisken1;
10 $islem_sonucu = ($degisken1 * 2) + $degisken2 + 6;
11 ?>

```

Şekil 3: Örnek PHP betiği: aritmetik işlemler.

Programın yapısını oluşturulan koşullar içinde karşılaştırmalar yapabilmek için yine benzer sembollerle ifade edilen koşul operatörleri kullanılır. Bunlar aşağıdaki Tablo 1'de gösterilmiştir.

Eşit	=
Eşit Değil	!=
Büyük	>
Küçük	<
Büyük Eşit	>=
Küçük Eşit	=<
Denk (veri türü de aynı)	===
Denk Değil (değeri veya veri türü farklı)	!==

Tablo 1: Koşul operatörleri.

PHP dilinde içerisinde birden fazla değer barındırabilen dizi (array) ismini verdiğimiz değişken türleri bulunmaktadır. Bunları değişken kümesi olarak ta düşünebiliriz. Bu tür değişkenleri tanımlamak için birden fazla yöntem bulunmaktadır. Şekil 4'te bulunan örnekte dizilerin farklı şekillerde nasıl

tanımlanacağı ve dizi içindeki değerlere erişimin nasıl yapılacağı görülebilir.

```
1 <?php
2 $dizi = array('bir', 'iki', 'üç');
3
4 $dizi[0] = 'bir';
5 $dizi[1] = 'iki';
6 $dizi[2] = 'üç';
7
8 echo "Dizinin ilk elemanı: " . $dizi[0];
9
10 echo "Dizinin tüm elemanları: ";
11
12 foreach ($dizi as $eleman) {
13     echo $eleman . " ";
14 }
15 ?>
```

Şekil 4: Örnek PHP betiği: dizilerin kullanılması ve dizi elemanlarına erişim.

Neseneye dayalı model ile birlikte gelen sınıflar ise "class" komutu ile tanımlanırlar.

PHP sürüm 5.3 ile birlikte nesneye dayalı model altyapısına "namespace" (isim uzayı) yapısı eklenmiştir. Böylelikle birinden farklı işlemler yapan fakat aynı isimleri alan sınıflar bir öbek altında toplanabilir hale getirilmiştir.

PHP'de nesneye dayalı model kapsamında "exception" (hata yakalama/ayıklama) kavramı ve "magic method" (sihirli yöntemler) bulunmaktadır. Sihirli yöntemler sınıfın oluşturulmasında, silinmesinde, özelliklerini atamada veya çağırma ve buna benzer birçok sınıf ile ilgili işlemlerde otomatik olarak tetiklenen ve çalıştırılan yöntemlerdir. Şekil 5'de sınıfların tanımlanması ve sihirli yöntemlerle ilgili örnek bir PHP betiği görülebilir.

```
1 <?php
2 class Ornek_Sinif {
3     public $isim = "Örnek Sınıf";
4
5     public function __construct() {
6         echo $this->isim . " oluşturuldu.";
7     }
8
9     public function isimGetir() {
10        return $this->isim;
11    }
12
13    public function __get($ozellik) {
14        if ($ozellik == "isim") {
15            return $this->isim;
16        }
17        else {
18            echo "Sınıf içinde $ozellik isimli bir özellik yok.";
19        }
20    }
21 }
22 ?>
```

Şekil 5: Örnek PHP betiği: sınıfların kullanılması ve sihirli yöntemler.

PHP dilinde nesneye dayalı model içinde görünürlük (visibility) ve kalıtım (inheritance) özelliği, özet (abstract) ve arayüz (interface) nesne türleri desteği bulunmaktadır.

Web programlamada tarayıcıdan girilen bilgilere erişim için "superglobal" ismi verilen ön tanımlı değişkenler bulunmaktadır. Sunucu, tarayıcı ve istek bilgilerine erişim için **\$_SERVER**, GET yönetimi ile gönderilen bilgiler için **\$_GET**, POST yöntemi ile gönderilen bilgiler için **\$_POST**, dosya yüklemesi sırasında yüklenen dosyaya ait bilgilere erişim için **\$_FILE**, çerez (cookie) bilgilerine erişim için **\$_COOKIE** ve sezon (session) bilgilerine erişim için ise **\$_SESSION** değişkeni kullanılır. Bu değişkenler PHP betiği içerisinde her yerden erişilebilir ve değerleri dizi olarak döner.

Dosya ve dizin işlemleri için C programlama dilinde kullanılan benzer isimlendirmeler ile fonksiyonlar tanımlanmıştır. Bunlara örnek olarak, "fopen", "fclose", "fgets" ve "feof" fonksiyonları verilebilir. Ek olarak dizin işlemleri için "opendir", "closedir" ve "readdir" gibi fonksiyonlar ve bu fonksiyonlara eşdeğer nesne arayüzleri bulunmaktadır.

PHP dilinde birçok işlemi basitleştirmek için yardımcı fonksiyonlar ve nesnelere bulunmaktadır. Bunlar kurulum sırasında veya kurulumdan sonra yüklenebilen eklentiler ile beraber gelirler. Eklentilerin sayısı yaklaşık 200'ü aşkındır.

Veritabanları ile çalışmak için PHP dilinde birçok eklenti vardır. Bunlardan en yaygın olanı MySQL veritabanı sunucusu için birden fazla eklenti ve diğer veritabanı sunucularını da destekleyen, PDO (PHP Database Object) ismi verilen bir ortak veritabanı arabirimi bulunmaktadır.

MySQL veritabanı sunucusu haricinde, PostgreSQL, SQLite, Oracle, MSSQL, Informix, FireBird/InterBase ve IBM DB2

gibi birçok veritabanı da eklentilerde desteklenmiştir. [6]

Son yıllarda sıklıkla karşımıza çıkan NoSQL [7] kavramı ile gelen ilişkisiz veritabanlarına örnek olarak MongoDB ve üçüncü parti Redis eklentisi, "key-value" (anahtar-değer) veritabanlarına örnek olarak Memcache için de eklentiler mevcuttur.

Bu eklentiler haricinde sıklıkla kullanılan;

- GD: Görüntü/resim işleme,
- SimpleXML: XML dosyalarını ve XML formatında metinleri işleme,
- COM: Microsoft Windows üzerinde COM nesnelere erişim,
- PCRE: Düzenli ifadeler,
- Mail: Elektronik posta gönderme,
- CURL: Uzak sunuculara HTTP, HTTPS ve FTP gibi protokoller ile erişim,
- APC: Önbellek işlemleri

eklentileri sayılabilir.

Eklenme geliştirmek için Zend Engine altında, C programlama dili ile hazırlanmış bir eklenti taslak kodu bulunmaktadır. Zend Engine ve PHP içinde bulunan eklentilerin kaynak kodları açık olduğu için bu eklentileri inceleyerek te ihtiyaç duyulan eklenti kolayca geliştirilebilir.

Kaynaklar

[1] Zend Technologies Ltd. PHP Web Application Server - PHP Development tools - PHP Training - Zend.com. <http://www.zend.com>

[2] The PHP Group. PHP 5.4 yenilikler. http://svn.php.net/viewvc/php/php-src/branches/PHP_5_4/NEWS?view=markup

[3] The PHP Group. PHP for Windows. <http://windows.php.net>

[4] Microsoft Corporation. PHP on IIS7: The Official Microsoft IIS Site. <http://php.iis.net>

[5] PHP Editor Review - PHP IDE. <http://www.php-editors.com>

[6] The PHP Group. PHP: İşlev Başvuru Kılavuzu - Manual. <http://www.php.net/manual/tr/funcref.php>

[7] Wikipedi. NoSQL (kavram) - Wikipedi. [http://tr.wikipedia.org/wiki/NoSQL_\(kavram\)](http://tr.wikipedia.org/wiki/NoSQL_(kavram))